

ベースにより掘削状況を判定する。

- (5) 異常があればその状況の説明と適切な対処方法をオペレータに提示する。
- (6) また、オペレータからの問い合わせに対し、推論過程の説明やセンサー情報、各種機器の状態、システムの状態等の表示を行う。
- (7) 掘削状況によって管理基準を変更することが多々ある。知識ベースと切り離して管理基準ベースを設けることにより、管理基準の変更を容易にした。

3. 知識表現

専門家の知識は次の3種類の形式を用いて表現した。

- (1) フレーム：フレーム型の知識はエキスパートシステムが対象とするプロセスの流れの中での変化のない情報、属性を表現する。言うなれば辞書を形成する。
 - (2) ルール：ルール型の知識は推論自体を表現する。また、プログラムの制御とオブジェクトの変化も表現する。
 - (3) オブジェクト：オブジェクト型の知識はプロセスの中で変わってくる情報を表わす。センサー情報や各種機器の状態、システムの状態等を表現する。
- これらの形式による表現例を図-2に示す。

4. リアルタイム処理の実現

リアルタイムエキスパートシステムが扱わなければならない

のは時間と共に変化するダイナミックなデータである。そのため、一般のエキスパートシステムで採用されているような推論機構をそのまま導入することはできない。①推論処理の高速化、②推論中のデータ変更の発生、③優先処理、④推論に時間がかかり過ぎる場合、といった問題の対応策が必要である。これらを解決するために次のような手段を施した。

- (1) ルールをグループ化して、使用する範囲を限定することにより推論実行の効率化を図った。

広くて浅い探索木を設計し、データとそのデータにとって意味のある探索木を定義した決定テーブルと呼ぶ対照表を用意してそれを基に、たどるべき探索木の範囲を限定して推論するようにした。このようにルールを索引付けることで、将来においてルールが増加しても処理速度の低下を招かなくて済む。

- (2) 複数の推論プロセスが発生した場合、優先度比較を行い、優先度の高いものを実行するようにした。

推論を階層化して、各階層内において異常レベルを3段階に分けた。そして、異常レベルの高いもの、推論階層の高いものほど高い優先権を与えた。

- (3) さらに、推論時間の管理を行い、一定の時間が経過したら強制的に推論を中断するようにした。

5. おわりに

我々は、本システムをEWS、P S 9000シリーズmodel 300のUNIX上で開発した。本システムには、100点ほどのアナログ信号とそれを上回る数のデジタル信号を2秒毎に取込み、処理しなければならないという、対話型のエキスパートシステムからは考えも及ばないほどの高速性が要求された。この課題を克服するために推論部はPrologで、それ以外の部分はCで記述した。なお、開発効率を高めるためにPrologにはCとの強力なインタフェースをもつIF/Prologを採用した。

今後、本システムをより多くの現場で適用し、さらに多くの適切な知識を獲得することによって幅広く運用できるシステムにしたいと考えている。

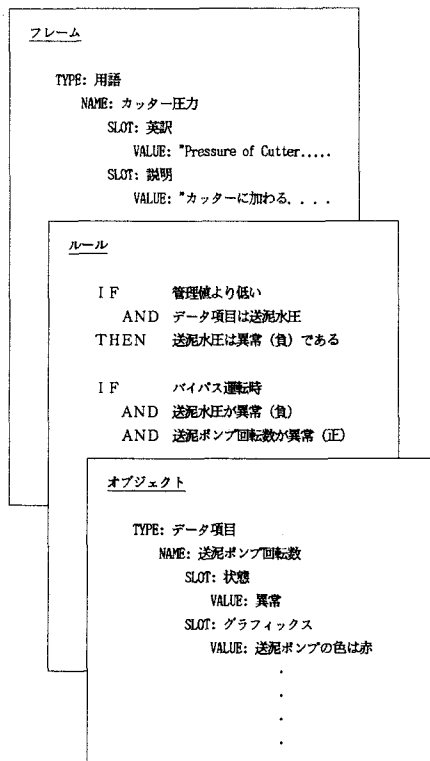


図-2 知識表現例